1. (Cancelled)

2. (Currently Amended) The method in claim ~~[1]~~ 6, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design;

for ones of said primary logical blocks that have a size above a predetermined maximum size limit, identifying secondary logical blocks of the second-highest level of said logical design hierarchy; and

iteratively repeating said process of identifying secondary logical blocks for additional levels of said logical design hierarchy until all logical blocks are within said predetermined maximum size limit.

3. (Original) The method in claim 2, further comprising calculating said predetermined maximum size limit by dividing the size of said integrated circuit design by the minimum number of logical blocks desired to be produced.

4. (Currently Amended) The method in claim ~~[1]~~ 6, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design; and

for ones of said primary logical blocks that have a size below a predetermined minimum size limit, combining said primary logical blocks until a combination of said primary logical blocks exceeds said predetermined minimum size limit.

5. (Original) The method in claim 4, further comprising calculating said predetermined minimum size limit by dividing the size of said integrated circuit design by the maximum number of logical blocks desired to be produced.

6. (Currently Amended) ~~The method in claim 1,~~ A method of partitioning an integrated circuit design comprising: identifying logical blocks within said integrated circuit design;

eliminating overlapping logical blocks that overlap above an overlap percentage limit; and

expanding remaining logical blocks to cover unused space within boundaries of said integrated circuit design,

wherein said process of eliminating overlapping logical blocks comprises: initially eliminating overlapping blocks based on said overlap percentage limit; counting the total number of remaining blocks; and revising said overlap percentage limit if said total number of remaining blocks is outside the range of the desired number of logical blocks.

7. (Currently Amended) The method in claim ~~[1]~~ 6, wherein said process of expanding said remaining logical blocks comprises:

expanding sides of said remaining blocks until said sides reach another block or reach a boundary of said integrated circuit design;

forming additional rectangles from remaining unused space; and

incorporating said additional rectangles into adjacent blocks.

8. (Cancelled).

9. (Currently Amended) The method in claim ~~[8]~~ 13, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design;

for ones of said primary logical blocks that have a size above said predetermined maximum size limit, identifying secondary logical blocks of the second-highest level of said logical design hierarchy; and

iteratively repeating said process of identifying secondary logic blocks for additional levels of said logical design hierarchy until all logical blocks are within said predetermined maximum size limit.

10. (Original) The method in claim 9, further comprising calculating said predetermined maximum size limit by dividing the size of said integrated circuit design by the minimum number of logical blocks desired to be produced.

11. (Currently Amended) The method in claim ~~[8]~~ 13, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design; and

for ones of said primary logical blocks that have a size below said predetermined minimum size limit, combining said primary logical blocks until a combination of said primary logical blocks exceeds said predetermined minimum size limit.

12. (Original) The method in claim 11, further comprising calculating said predetermined minimum size limit by dividing the size of said integrated circuit design by the maximum number of logical blocks desired to be produced.

13. (Currently Amended) ~~The method in claim 8,~~ A method of partitioning an integrated circuit design comprising: identifying logical blocks within said integrated circuit design, wherein said logical blocks are within a predetermined maximum size limit and a predetermined minimum size limit;

eliminating overlapping logical blocks that overlap above an overlap percentage limit;

expanding remaining logical blocks to cover unused space within boundaries of said integrated circuit design;

partitioning said integrated circuit design into partitions corresponding to said remaining

blocks; and

running applications within each partition of said integrated circuit design in parallel,

wherein said process of eliminating overlapping logical blocks comprises:

initially eliminating overlapping blocks based on said overlap percentage limit;

counting the total number of remaining blocks; and

revising said overlap percentage limit if said total number of remaining blocks is outside the range of the desired number of logical blocks.

14. (Currently Amended) The method in claim ~~[8]~~ 13, wherein said process of expanding said remaining logical blocks comprises:

expanding sides of said remaining blocks until said sides reach another block or reach a boundary of said integrated circuit design;

forming additional rectangles from remaining unused space; and

incorporating said additional rectangles into adjacent blocks.

15. (Cancelled).

16. (Currently Amended) The method in claim ~~[15]~~ 20, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design;

for ones of said primary logical blocks that have a size above a predetermined maximum size limit, identifying secondary logical blocks of the second-highest level of said logical design hierarchy; and

iteratively repeating said process of identifying secondary logic blocks for additional levels of said logical design hierarchy until all logical blocks are within said predetermined maximum size limit.

17. (Original) The method in claim 16, further comprising calculating said predetermined maximum size limit by dividing the size of said integrated circuit design by the minimum number of logical blocks desired to be produced.

18. (Currently Amended) The method in claim ~~[15]~~ 20, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design; and

for ones of said primary logical blocks that have a size below a predetermined minimum size limit, combining said primary logical blocks until a combination of said primary logical blocks exceeds said predetermined minimum size limit.

19. (Original) The method in claim 18, further comprising calculating said predetermined minimum size limit by dividing the size of said integrated circuit design by the maximum number of logical blocks desired to be produced.

20. (Currently Amended) ~~The method in claim 15,~~ A method of partitioning an integrated circuit design comprising:

identifying logical blocks within said integrated circuit design;

eliminating overlapping logical blocks that overlap above an overlap percentage limit;

expanding remaining logical blocks to cover unused space within boundaries of said integrated circuit design by expanding sides of said remaining blocks until said sides reach another block or reach a boundary of said integrated circuit design;

forming additional rectangles from remaining unused space; and

incorporating said additional rectangles into adjacent blocks.

wherein said process of eliminating overlapping logical blocks comprises:

initially eliminating overlapping blocks based on said overlap percentage limit;

counting the total number of remaining blocks; and

revising said overlap percentage limit if said total number of remaining blocks is outside the range of the desired number of logical blocks.

21. (Cancelled).

22. (Currently Amended) The program storage device in claim ~~[21]~~ 26, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design;

for ones of said primary logical blocks that have a size above a predetermined maximum size limit, identifying secondary logical blocks of the second-highest level of said logical design hierarchy; and

iteratively repeating said process of identifying secondary logic blocks for additional levels of said logical design hierarchy until all logical blocks are within said predetermined maximum size limit.

23. (Original) The program storage device in claim 22, wherein said method further comprises calculating said predetermined maximum size limit by dividing the size of said integrated circuit design by the minimum number of logical blocks desired to be produced.

24. (Currently Amended) The program storage device in claim ~~[21]~~ 26, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design; and

for ones of said primary logical blocks that have a size below a predetermined minimum size limit, combining said primary logical blocks until a combination of said primary logical blocks exceeds said predetermined minimum size limit.

25. (Original) The program storage device in claim 24, wherein said method further comprises calculating said predetermined minimum size limit by dividing the size of said integrated circuit design by the maximum number of logical blocks desired to be produced.

26. (Currently Amended) ~~The program storage device in claim 21,~~ A program storage device for use with a computer, said program storage device tangibly embodying a program of instructions executable by said computer for performing a method of partitioning an integrated circuit design, said method comprising:

identifying logical blocks within said integrated circuit design;

eliminating overlapping logical blocks that overlap above an overlap percentage limit; and

expanding remaining logical blocks to cover unused space within boundaries of said integrated circuit design,

wherein said process of eliminating overlapping logical blocks comprises:

initially eliminating overlapping blocks based on said overlap percentage limit;

counting the total number of remaining blocks; and

revising said overlap percentage limit if said total number of remaining blocks is outside the range of the desired number of logical blocks.

27. (Currently Amended) The program storage device in claim ~~[21]~~ 26, wherein said process of expanding said remaining blocks comprises:

expanding sides of said remaining blocks until said sides reach another block or reach a boundary of said integrated circuit design;

forming additional rectangles from remaining unused space; and

incorporating said additional rectangles into adjacent blocks.

28. (Cancelled).

29. (Currently Amended) The service in claim ~~[28]~~ 33, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design;

for ones of said primary logical blocks that have a size above a predetermined maximum size limit, identifying secondary logical blocks of the second-highest level of said logical design hierarchy; and

iteratively repeating said process of identifying secondary logic blocks for additional levels of said logical design hierarchy until all logical blocks are within said predetermined maximum size limit.

30. (Original) The service in claim 29, further comprising calculating said predetermined maximum size limit by dividing the size of said integrated circuit design by the minimum number of logical blocks desired to be produced.

31. (Currently Amended) The service in claim ~~[28]~~ 33, wherein said process of identifying logical blocks comprises:

identifying primary logical blocks comprising the highest level of logical design hierarchy of said integrated circuit design; and

for ones of said primary logical blocks that have a size below a predetermined minimum size limit, combining said primary logical blocks until a combination of said primary logical blocks exceeds said predetermined minimum size limit.

32. (Original) The service in claim 31, further comprising calculating said predetermined

minimum size limit by dividing the size of said integrated circuit design by the maximum number of logical blocks desired to be produced.

33. (Currently Amended) ~~The service in claim 28,~~ A service of partitioning an integrated circuit design comprising:

identifying logical blocks within said integrated circuit design;

eliminating overlapping logical blocks that overlap above an overlap percentage limit; and

expanding remaining logical blocks to cover unused space within boundaries of said integrated circuit design.

wherein said process of eliminating overlapping logical blocks comprises:

initially eliminating overlapping blocks based on said overlap percentage limit;

counting the total number of remaining blocks; and revising said overlap percentage limit if said total number of remaining blocks is outside the range of the desired number of logical blocks.

34. (Currently Amended) The service in claim ~~[28]~~ 33, wherein said process of expanding said remaining logical blocks comprises:

expanding sides of said remaining blocks until said sides reach another block or reach a boundary of said integrated circuit design;

forming additional rectangles from remaining unused space; and

incorporating said additional rectangles into adjacent blocks.

35. (Cancelled).